Name: Pearl Law
SCU ID: W0958839
Rank: 1
mAP score: 0.7091

Multi-class Image Object Detection with YOLOv5

Abstract

The goal of this project is to develop an object detection model from an existing detection network model that can determine the class and location of one or more object within an image. Transfer learning was applied using the YOLOv5 neural network originally trained on COCO128 dataset to generate a predictive model that can detect seven specific types of objects. Inference was performed on the test dataset using the best weights obtained during transfer learning, which yielded mean average precision (mAP) of 0.7091.

Introduction

Transfer learning is a useful tool that allows one network to be built from an existing trained network. By utilizing another pre-trained network's weights, we can speed up training time for similar datasets while maintaining desirable task performance. This project utilizes the YOLOv5 network model to train a neural network adept at object detection for seven different types of objects (small truck, medium truck, large truck, bus, van, SUV, or car) within a given image. The entire training dataset consists of 15000 images and a text file containing bounding box labels for all training images. The validation dataset consists of 2000 images and a text file containing bounding box labels for all validation images. There are 2000 test images to perform inference on to measure the performance of the pre-trained YOLOv5 model at object detection. This study evaluates the application of transfer learning and the effects of tuning key hyperparameters (i.e. learning rate and weight decay) on developing a predictive model that can accurately determine the class and location of seven object types.

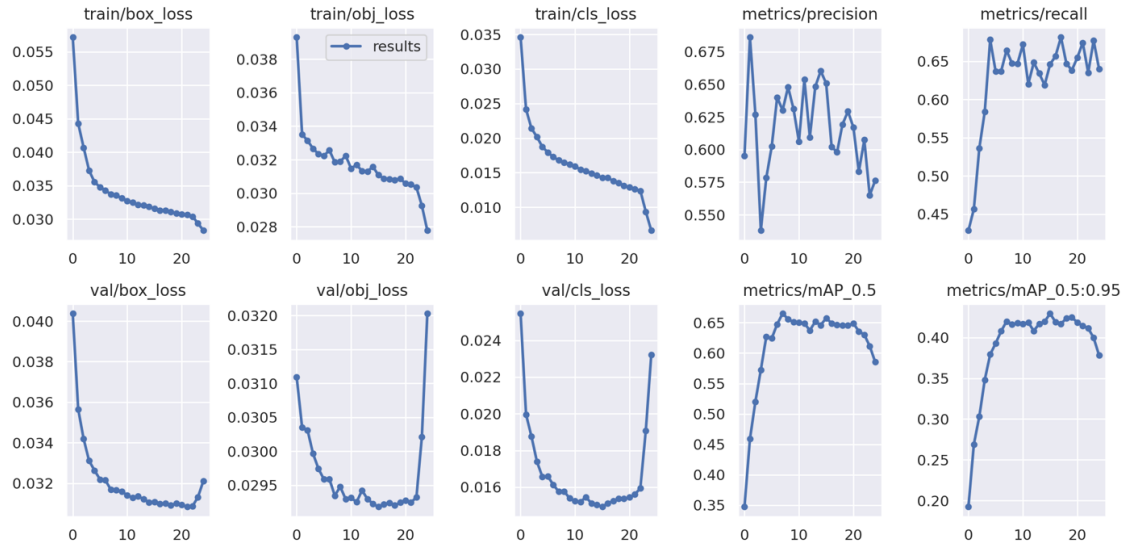
Approach/Methods

The YOLOv5 model requires an individual text file containing every bounding box label for each image. This was accomplished by matching each image file name with the image id in the labels.txt file and extracting all bounding boxes corresponding to an image into a single text file. Each line in a text file corresponds to one bounding box and is formatted as <class_number> <center_x> <center_y> <width> <height>, where class numbers are zero-indexed and box coordinates are normalized between the range 0 to 1, inclusive. The result of this preprocessing step generated 15000 training label text files corresponding to the existing 15000 training image files and 2000 validation label text files corresponding to the existing 2000 validation image files. Each image was resized to 640 x 640 to match the default input size for YOLOv5. The specific model chosen for this project was YOLOv5l, which is the second largest network of all YOLOv5 models and contains 47.8M total parameters. While YOLOv5l takes longer to train compared to YOLOv5s and YOLOv5m due to its complexity, it results in much higher mAP than the smaller, simpler models.

Results

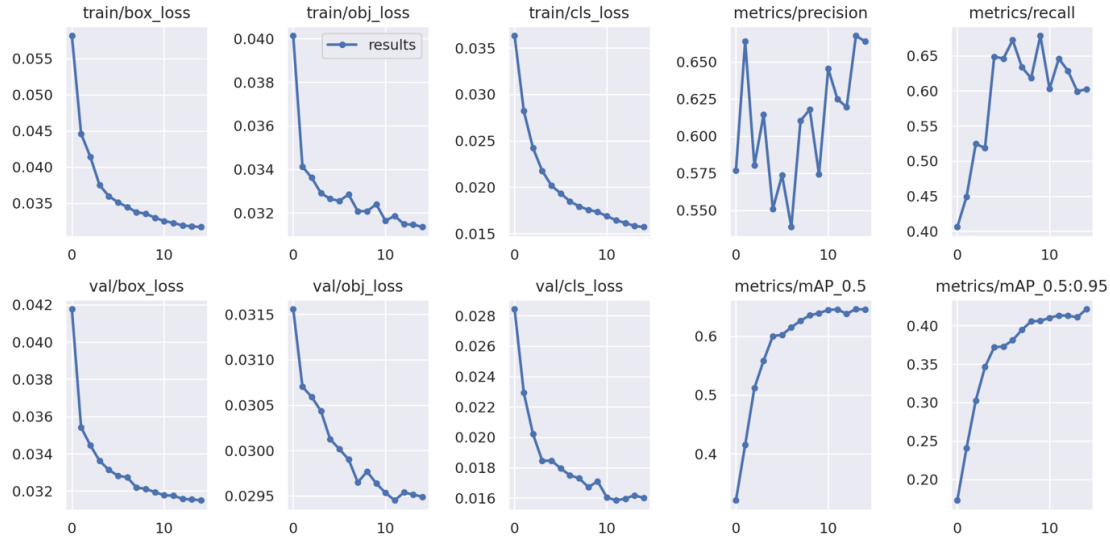
Training was performed under three separate scenarios and optimized with stochastic gradient descent (momentum = 0.937):

1. Training with the default pre-trained YOLOv5l model



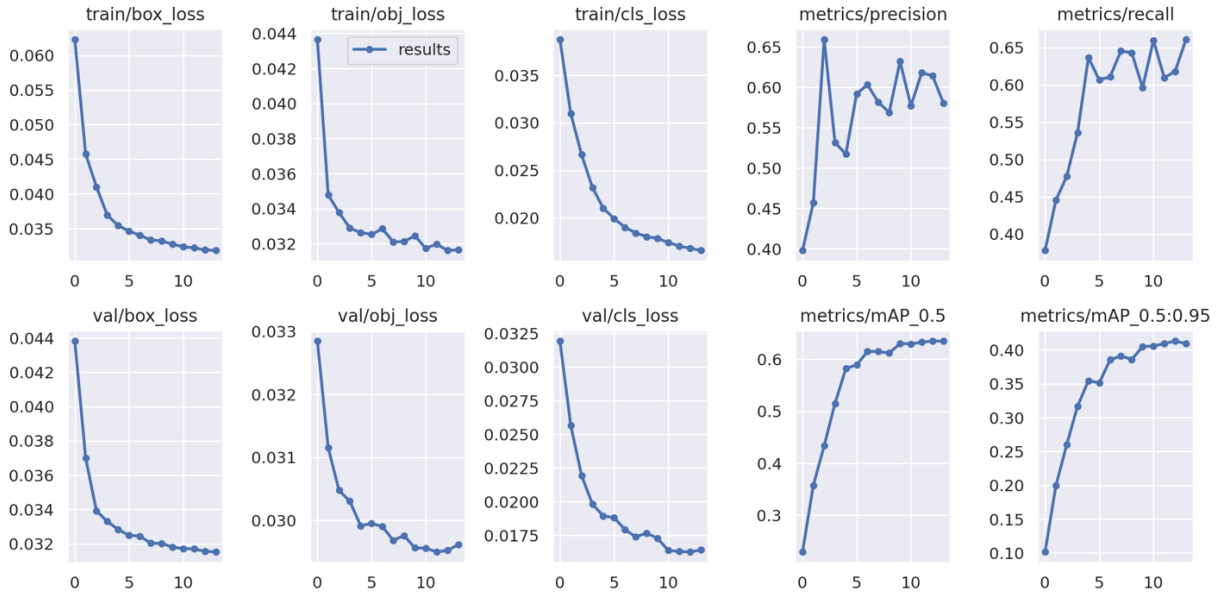
- Batch size = 16, epochs = 25, learning rate = 0.01, weight decay = 0.0005
- Results for best epoch (15):
  - mAP@0.5 = 0.6577
  - mAP@0.5:0.95 = 0.4296
  - Precision = 0.6508
  - Recall = 0.6461

2. Training with the first nine layers (backbone) of the model frozen and all others unrestricted from learning (transfer learning protocol)



- Batch size = 16, epochs = 15, learning rate = 0.01, weight decay = 0.0005
- Results for best epoch (14):
    - mAP@0.5 = 0.6451
    - mAP@0.5:0.95 = 0.4214
    - Precision = 0.6637
    - Recall = 0.6024

3. Transfer learning protocol with fine-tuning of hyperparameters



- Batch size = 16, epochs = 14, learning rate = 0.003, weight decay = 0.0003
- Results for best epoch (12):
    - mAP@0.5 = 0.6351
    - mAP@0.5:0.95 = 0.4134
    - Precision = 0.6141
    - Recall = 0.6182

The optimal weights derived from the transfer learning protocol were obtained and used for inference. Object confidence and IoU threshold values were varied to test for variance and bias in object detection at different threshold values. The best object confidence threshold (0.25) and IoU threshold (0.45) values demonstrating greatest variance and lowest bias in the model was used for inference.

Conclusion

Previous research demonstrated that the YOLOv5 model outperforms earlier YOLO models in terms of accuracy at the expense of speed. Since the goal of this project was to achieve the highest possible mAP, YOLOv5 was selected over the other YOLO models for transfer learning. Applying the optimal weights from training via transfer learning with the YOLOv5l model to detect seven different object types and their locations in an image generated an mAP of 0.7091. This indicates that the pre-trained neural network was able to generalize well to this specific dataset and perform the object detection tasks fairly accurately.